

# 南工基礎Python教學--week01

## 1. 建築科為什麼要學程式、懂電腦？

(我自己本身對於建築方面的知識為 0，所以這些知識也是我問 GPT 的，不要吉我 Q Q)

### 1.1. 在自動化方面

AutoCAD、SketchUp、Revit 這些工具大家都會用 → 你要更有競爭力，就要會做「自動化」的東西。

例子：你一天畫 20 張圖，如果你會寫一點程式，自動產生、標註、轉格式，別人做 5 小時，你半小時搞定。

別誤會，這邊的【自動化】，是指【一些重複性且無差異的動作】可以透過【腳本】的方式去執行。

相關文章：

- <https://allaboutdataanalysis.medium.com/10-個令人驚歎的-python-自動化腳本-23e57eb4e469>
- <https://www.youtube.com/watch?v=mYX9AaJF2lw>

### 1.2. 不懂科技，進工地會一臉茫然

「你不需要會寫很複雜的程式，但你要看得懂、知道怎麼配合做事。」

現在很多工地用 BIM、3D 建模、雲端協作平台，圖不是印出來，是大家一起線上看。

一些建材計算、進度排程，背後也跑程式邏輯。

如果你只會看平面圖、不會操作系統，會很容易變成「需要人家教你怎麼做的人」。

### 1.3. 數位雙生 (Digital Twin)

【讓你設計的建築「有一個數位的分身」】

什麼是數位雙生？：指在電腦裡建一個跟現實建築一模一樣的「虛擬模型」，可以即時監控、模擬、預測真實建築的狀況。

比 BIM 更進一步：它不只是設計，而是營運中也能持續更新的活資料模型。



## 1.4. 監控鏡頭與感測器

建築中會裝很多感測器和鏡頭，不只是安全，還有很多管理用處：

- 偵測人流 → 控制冷氣強度
- 錄影紀錄 → 工地責任歸屬（事故追蹤）
- 辨識車牌、門禁 → 出入控管自動化



## 2. thonny IDE

【我們要用 python 寫程式】，那這句話當中，你可能有兩個名詞不太懂，Python 是甚麼？程式是甚麼？



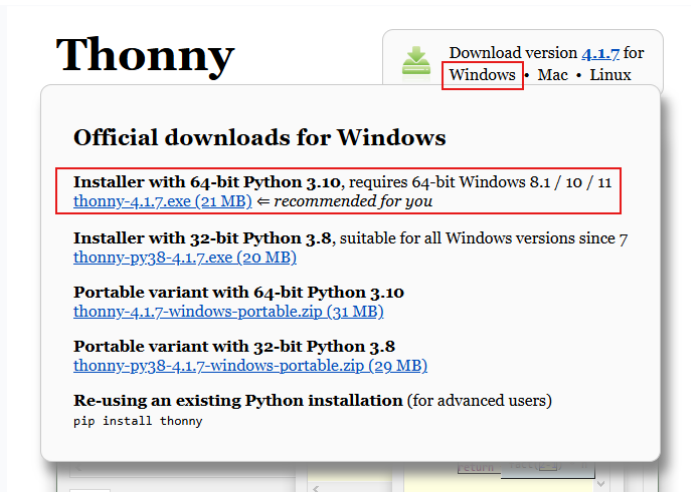
這樣說好了，假設我們要煮飯：

- 煮飯是一個動作(就是寫程式)
- 然後我們會需要白米(就是 python)
- 我們需要電鍋(這個就是 IDE)
  - 煮飯可以有很多方式，可以用電鍋，可以自己炊 etc，IDE 也是依樣。

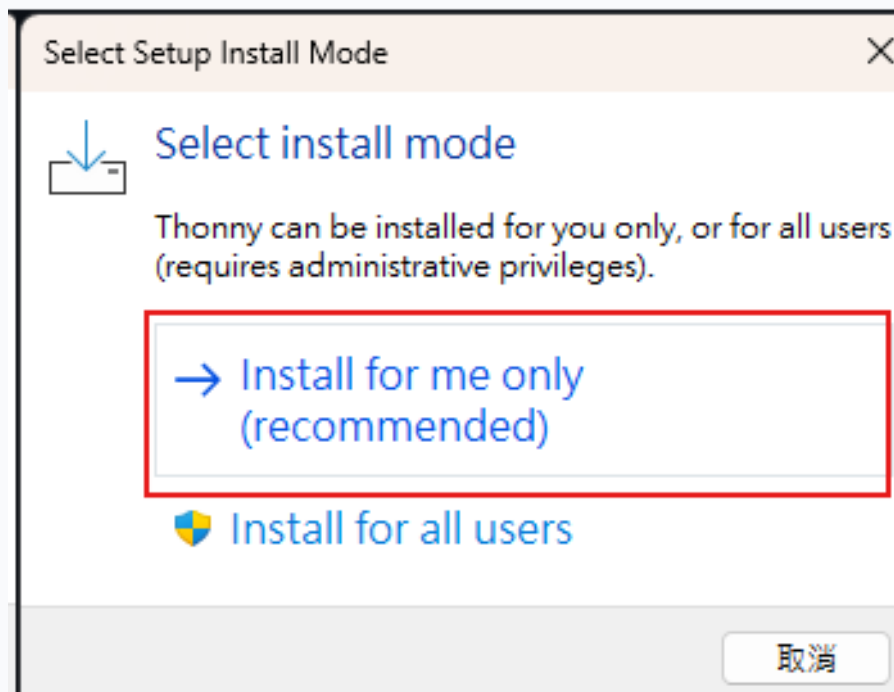
這次這次我們選的電鍋(IDE)就是 thonny

### 2.1. 安裝 thonny

安裝連結：[Thonny, Python IDE for beginners](#)



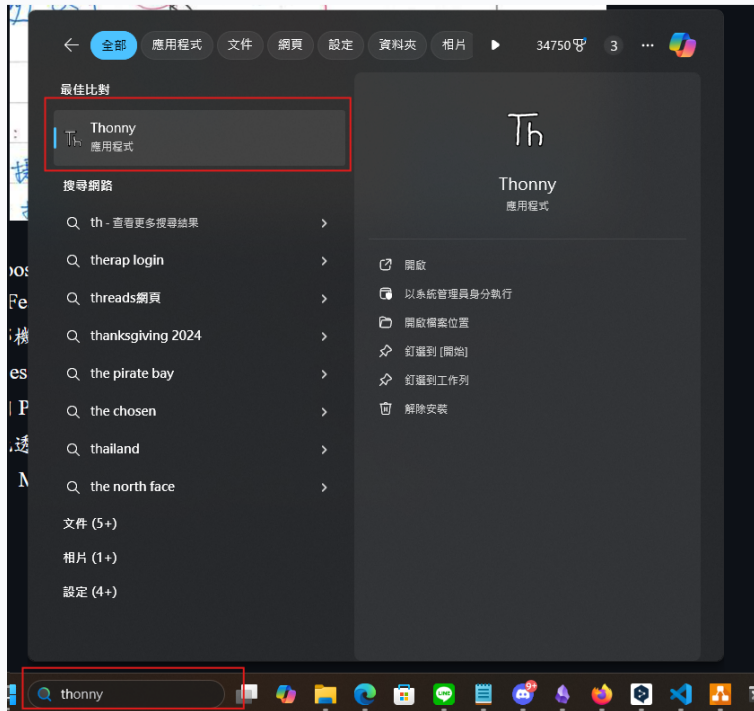
就是一直預設安裝即可(簡單八)



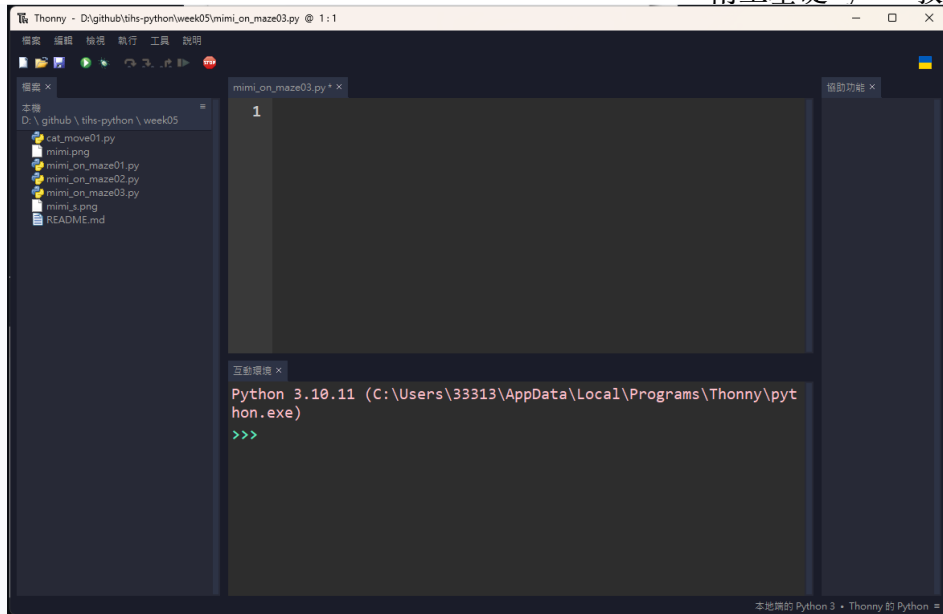
這樣就安裝好摟！讓我們來啟動她。



## 2.2. 使用 thonny

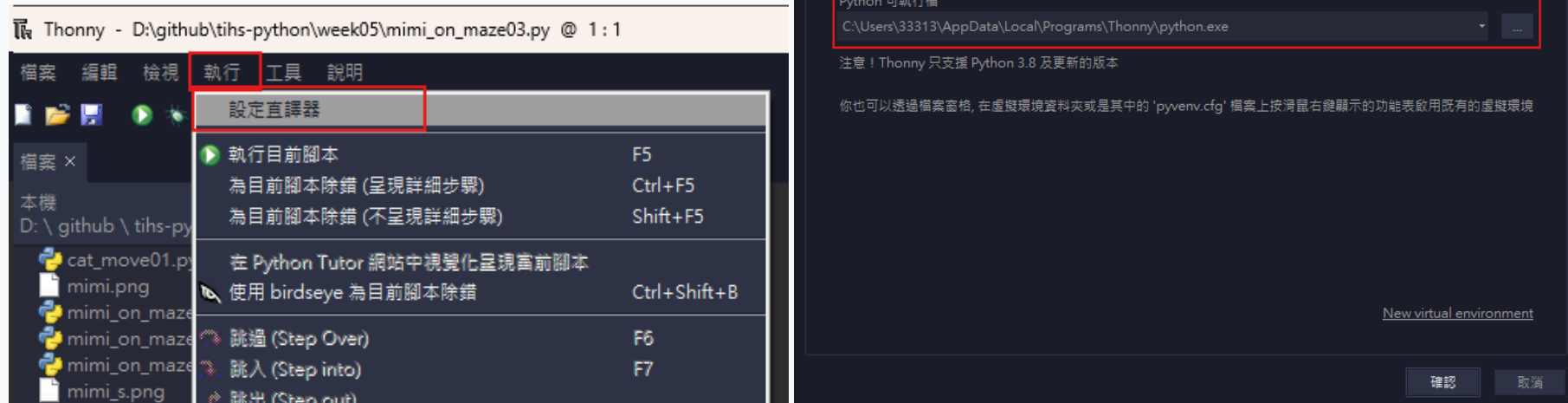


在打開後，他會長這樣(可能顏色會不同，但基本版面都是這樣)



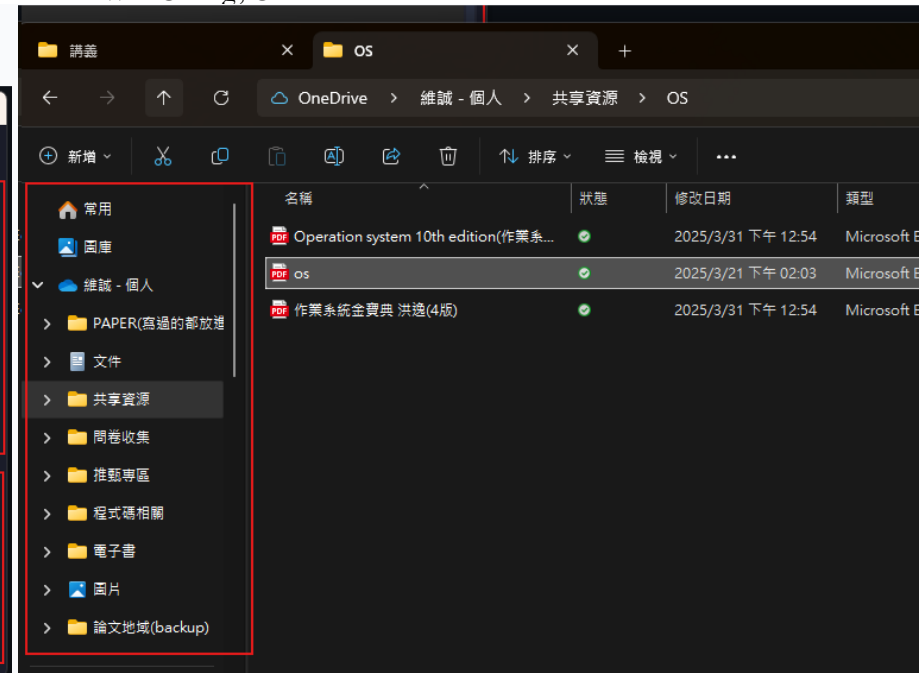
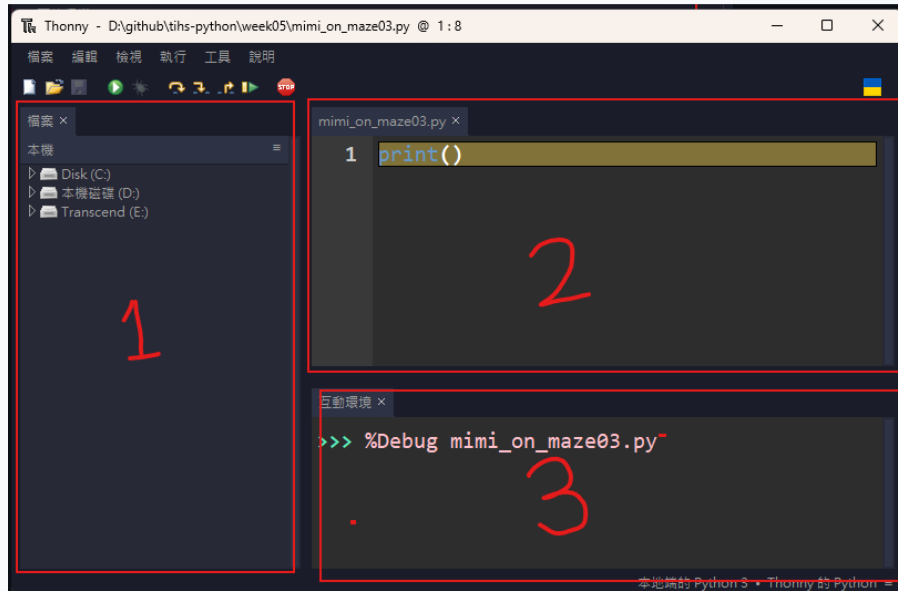
首先，我們要對電鍋做一些設定參數

IDE 他本身是一個編輯器環境，我們要告訴他，要用【python】寫程式



我們來簡單介紹一下這個 IDE 版面跟操作：

- 1 號區塊就是檔案路徑，就跟你的檔案總管一樣。
- 2 號區塊就是讓你寫程式的地方。
- 3 號區塊就是程式執行，顯示結果的地方。

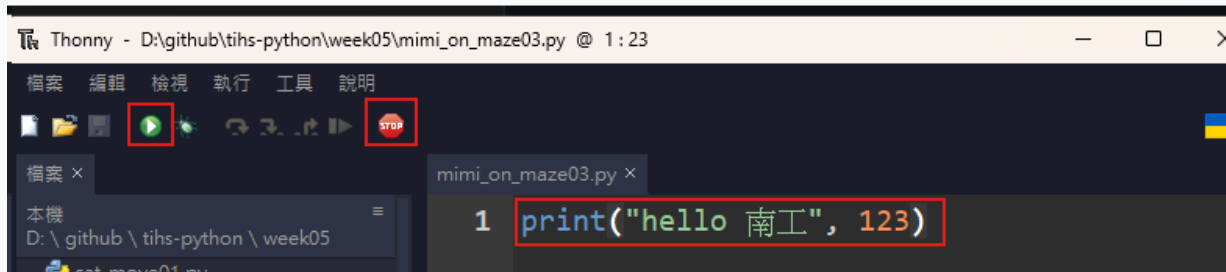


那接下來幫我在【2 號區塊】寫下這一行。

```
print("hello 南工", 123)
```

要讓程式運行，就按綠色按鈕(要記得儲存(ctrl+S))

要讓程式停止，就按紅色按鈕





```
互動環境 x  
hello 南工 123  
>>>
```

到這邊的話，代表你們安裝完成並成功執行程式了(對就是這樣 XD



#### ✎ 補充

這部動畫較【Re:從零開始的異世界生活】  
超級好看!!!

### 3. 基本輸出 print

學習每一個程式的第一步，就是要將你的程式碼給【印出來】  
(讓你有種， 歐齁~電腦在聽我的話耶~ 這種感覺)

如何將程式碼運行出來呢？

```
# print(妳想要輸出的東西)
```

```
print(1+2)  
print(3*5)  
print(7/2)
```

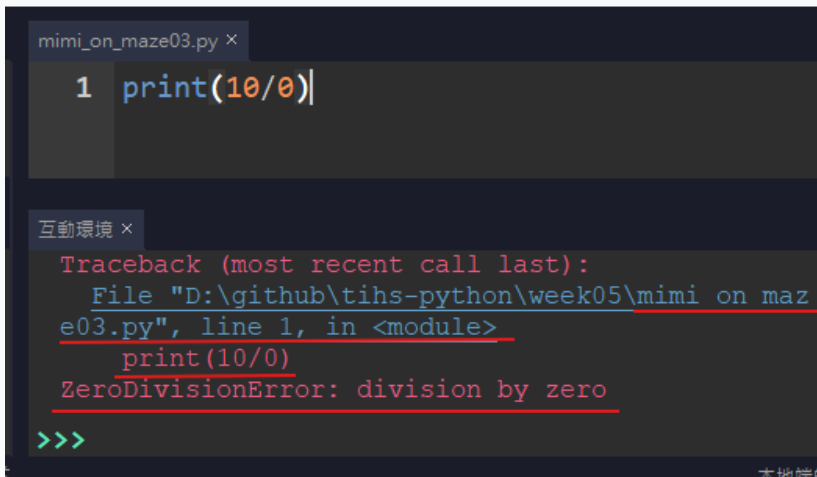
```
print(7%2)
print("助教好可愛")
```

#### ④ 補充：

整數(int)、浮點數(float)

那如果我來做一些奇怪的事怎麼樣，像是～

```
print(10/0)
```

A screenshot of a Python IDE window titled 'mimi\_on\_maze03.py'. The code editor shows a single line of code: '1 print(10/0)'. Below the code editor, the '互動環境' (Interactive Environment) pane displays a traceback. The traceback text is: 'Traceback (most recent call last):  
File "D:\github\tihs-python\week05\mimi on maze03.py", line 1, in <module>  
 print(10/0)  
ZeroDivisionError: division by zero'. The prompt '>>>' is visible at the bottom of the interactive environment.

```
mimi_on_maze03.py ×
1 print(10/0)

互動環境 ×
Traceback (most recent call last):
  File "D:\github\tihs-python\week05\mimi on maze03.py", line 1, in <module>
    print(10/0)
ZeroDivisionError: division by zero
>>>
```

#### 🔗 他報錯了

上一次看到紅色的東西，應該是校門口的紅綠燈對吧  
紅色通常就是告訴你，遇到了一些錯誤，請你去把這個錯誤處理掉！！

## 3.1. 嘗試輸出字串

我們來試試看輸出「文字」（我們以後會叫他字串（string））

```
# error
你好，我是助教，今天很高興認識你

# correct
"你好，我是助教，今天很高興認識你"

# 可以用單引號或雙引號
```

## 4. 變數

那其實，我們盡可能的會把我們要的東西，用一個【變數】去代表他  
假設我今天要輸出的內容超級長，我總不能讓費時間在打內容本身吧 XD



```
# bad
print("從零開始的異世界生活")
print("從零開始的異世界生活")

# good
t = "從零開始的異世界生活"
print(t)
```

## 4.1. 變數命名方式

這很重要，名子取不好的話，助教就會氣到發光歐



```
# correct
x = 0
gold = 1000
TA = "助教好可愛"
TA_cute = "助教好可愛"

# error
1play = 0 # 數字不可以開頭
@play = 0 # 特殊文字不可以開頭
if = 0 # 保留字不可以使用
```

## 5. 資料類型(data type)

在 Python 裡，每一個變數，都有一個屬性：資料類型

就像每個角色都有職業一樣（劍士、魔法師、補師），不同的資料型態，能做的事情也不一樣！

資料型態	範例	說明
int（整數）	123, -7, 0	不帶小數點的數字
float（浮點）	3.14, -0.5	有小數點的數字

資料型態	範例	說明
str (字串)	"雷姆" 、 '123'	用引號包住的文字
bool (布林)	True , False	只有兩種值的邏輯判斷

可以透過 `type()`得知某個變數的類型。

```
a = 123
b = 3.14
c = "雷姆好可愛"
d = True
```

```
print(type(a)) # int
print(type(b)) # float
print(type(c)) # str
print(type(d)) # bool
```

```
# 文字型的數字 ≠ 數值
x = "123" # 字串
y = 123   # 整數
```

```
print(x + x) # 結果是 "123123"
print(y + y) # 結果是 246
```

```
# NoneType 表示 什麼都沒有、空空的 (等於 null)
# 常用在「還沒決定值」的狀況 (像剛創角色還沒選職業)
x = None
print(type(x)) # <class 'NoneType'>
```

≡ Lab：試著建立以下變數，並用 `type()` 印出資料型態

```
num = 2025
pi = 3.14159
name = "艾姬多娜"
```

```
is_cute = True  
nothing = None
```

## 6. 輸入資料(input)

程序在執行時，會有【等待使用者】輸入的時機  
我們要如何時做呢

```
n = input("這邊可以輸入你要的資料：")  
print(n)  
print(type(n))  
  
# 這邊可以輸入你要的資料：45  
# 45  
# <class 'str'>
```

這邊可以發現，我在讀取之後，讀取進來的資料是【string(字串)】(是不能做加減計算的)  
如果我要的是，可以計算的【數值】=>【integer(整數)】

```
n = int(input("這邊可以輸入你要的資料："))  
print(n)  
print(type(n))  
  
# 這邊可以輸入你要的資料：0123  
# 123  
# <class 'int'>
```

上述是提到【字串轉整數】，如果是【整數轉字串】呢

```
n = int(input("這邊可以輸入你要的資料："))  
print(type(n))  
m = str(n)  
print(m)
```

```
print(type(m))
```

```
# 這邊可以輸入你要的資料：16
```

```
# <class 'int'>
```

```
# 16
```

```
# <class 'str'>
```